# *Index-Based Symmetric DNA Encryption Algorithm*

Zhang Yunpeng
School of Software and Microelectronics
Northwestern Polytechnical University
Xi’an, Shannxi, 710072, P.R.China
Melbourne eResearch Group
The University of Melbourne
Barry St,PARKVILLE,Victoria,3010j,Australia
poweryp@163.com

Zhu Yu
School of Software and Microelectronics
Northwestern Polytechnical University
Xi’an, Shannxi, 710072, P.R.China

Wang Zhong
School of Software and Microelectronics
Northwestern Polytechnical University
Xi’an, Shannxi, 710072, P.R.China
zhongw1115@126.com

Richard O.Sinnott
Melbourne eResearch Group
The University of Melbourne
Barry St, PARKVILLE, Victoria 3010 Australia
rsinnott@unimelb.edu.cn

***Abstract*—In this paper, a new index-based symmetric DNA encryption algorithm has been proposed. Adopting the methods of Block-Cipher and Index of string, the algorithm encrypts the DNA-sequence-based plaintext. First, the algorithm encodes each character into ASCII codes. And then, according to the nucleotide sequence, the researcher should convert it to the DNA coding. Besides, the researcher selects the special DNA sequence as the encryption index, and likewise, the pretreated plaintext will be divided into different groups. Next, the key created by the Chaos Key Generator based on the Logistic Mapping and initialized by the number $x_0$ and $\mu$, will take XOR operation with the block-plaintext. The type of number $x_0$ and $\mu$, which is selected by the researcher, is double. Then, the result of these processes will be translated on the DNA sequence. In addition, compared to special DNA sequence, the algorithm finds the sequence which has no difference with it. Then, the algorithm will store the position as the Cipher-text. The researcher proves the validity of the algorithm through simulation and the theoretical analysis, including bio-security and math-security. The algorithm has a huge key space, high sensitivity to plaintext, and an extremely great effect on encryption. Also, it has been proved that the algorithm has achieved the computing-security level in the encryption security estimating system.***

***Keywords-DNA Cryptography, Index, Block Encryption, Chaos***

## I. INTRODUCTION

With the rapid development of modern communications technology and the Internet, the importance of information security becomes highlighted. DNA cryptology has progressed been put forward as a newly technique, which, together with traditional quantum cryptology, formed the three main branches of cryptology. DNA Encryption means combing DNA technique with cryptology, producing new cryptography to provide safe and efficient cipher services. The difference between DNA cryptography and traditional one is that the former is based on the limitation of biotechnology, which is unrelated to numeracy. Thus, it is immune to the attack from super computer. Certainly the security also needs serious mathematical justification. But it is undeniable that with the further development of biotechnology and cryptology, DNA’s vast parallelism, extraordinary information density and exceptional energy efficiency could make large-scale data storage and encryption or decryption faster. When the information is encrypted which is less demanding for parallel real-time data or extremely demanding for mass data storage applications, it has a unique advantage. Index-based DNA symmetric encryption methods, definitely belongs to this field, is presented in this paper. The searcher proves that it is an efficient, sage, practical, encrypting way, and also it can efficiently prevent attacks.

## II. SYMMETRIC-CRYPTOGRAPHY AND DNA-CRYPTOGRAPHY

### 2.1. Symmetric Cryptography

Cryptography is constructed by five elements { M，C，K，E，D },among which Message space M is also called Plaintext space, Cipher text space C, Key space K, Encryption algorithm E and Decryption algorithm D.

For each plaintext m in Message space M, Encryption algorithm E can encrypt the m to the Cipher text c with the secret key of $k_e$; and the Decryption algorithm D could decrypt the Cipher text c to the plaintext m with the key of $k_d$.

The process of encryption could be expressed as:

$$E_{Ke}\left(P\right)=C \tag{1}$$

The process of decryption could be expressed as:

$$D_{Kd}\left(C\right)=P \tag{2}$$

According to Key’s feature, Cryptography can be divided into Symmetric Cryptosystem and Asymmetric Cryptosystem. Based on the different methods of encryption, Symmetric

Cryptosystem falls into Stream Cipher and Block Cipher. In the field of Stream Cipher, in order to get cipher text, the researcher needs to compute the plaintext by-bit with the word created by Pseudorandom bit generator. However, in the field of Block Cipher, researcher should divide plaintext into different groups, and then encrypt each group. The approach mentioned in this paper adopts the idea of Block Cipher, in the organization of Asymmetric Cryptosystem based on the Stream Cipher. In addition, because of the security of Stream Cipher largely depending on the Key sequence, we apply a Chaos Key Sequence Generator to create a well organized random key sequence to ensure the safety of Cryptography mentioned in this paper.

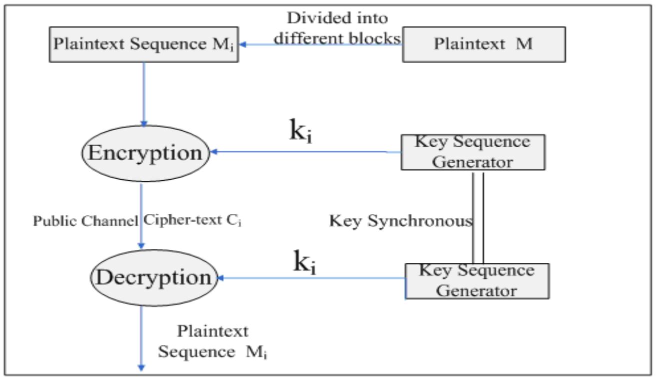

**Figure 1.** The Encryption Communication Model

### 2.2. DNA Cryptography

DNA, as an information carrier, has an extraordinary storage density, which can better solve the problem of how to create or store the large quantity of Pad. DNA Cryptography is implemented by using modern biological techniques as tools and DNA as information carrier to fully exert the inherent advantages of high storage density and high parallelism to achieve encryption.

Based on the DNA double helix structure and the principle of Watson-Crick complementary, the encryption and decryption can be simulated as the DNA's biological or chemical process. It encodes the information and then stores plaintext, cipher-text or other information on DNA encoding nucleotide sequence after significant operation. By now, the password system has been built.

In recent years, researchers have proposed a large number of DNA-based encryption algorithms, as a result of that DNA cryptography is still in the initial stage, it don't have a complete model and an efficient verification mechanism. For example, Gehani[1] proposed a one-pad encryption based on the DNA encryption and decryption methods; Leier[2] designed two kinds of encryption scheme to fulfill information hiding based on DNA binary sequences; applying DNA technology, Kazuo[3] solved the problem of distributing key, etc.[4-5]。 At the same time, DNA encryption system based on symmetric encryption has also made progress. In 2006, Sherif T. Amin[6] proposed YAEA encryption algorithm is essentially a symmetric encryption algorithm based on DNA; Kang Ning[3] presented that primer together as key can build DNA encryption algorithm. Both are symmetric encryption algorithms.

## III. INDEX-BASED SYMMETRIC DNA ENCRYPTION CRIPTION OF ALGORITHM

### 3.1. Coding Method

In order to make the plaintext can be calculated with the key and expressed in DNA sequence, the encryption method presented in this paper requires researcher to encrypt the plaintext twice. The specific steps are as following:

1. Translate the plaintext into ASCII code.

Encoding the plaintext as ASCII characters, so it could be expressed as many 8-bit binary numbers. For example, when we want to encode plaintext 'H', based on the table of ASCII code, we could find that ASCII $[H]_{10}$ = 72. Therefore, we express character 'H' in Binary: 'H' = $(72)_2$= 01001000.

2. Encoding the plaintext as DNA sequence.

Similar to the traditional encryption, the algorithm's efficiency of DNA encryption directly affects the encoding quality. In the literature [7], Adleman and Lipton use three bits to represent a base character randomly in their own researches. The basic variety of code which is based on 4 kinds of bases is $4^3$=64, enough to meet code requirements. However, this encoding is not reasonable. The algorithm presented in this paper applies a more optimal encoding formula (3) to express the plaintext on DNA sequence [8].

$$\begin{aligned} 00 &\rightarrow C \\ 01 &\rightarrow T \\ 10 &\rightarrow A \\ 11 &\rightarrow G \end{aligned} \tag{3}$$

For example, the binary coding of the character 'H' is 01 00 10 00. So, through DNA encoding, the character 'H' could be expressed on the DNA sequence as "TCAC". To sum up, through the encoding process presented above, the plaintext will be expressed on the DNA sequence in the form of the nucleotide.

The method of decryption shares the same process.

### 3.2. Key Select and the Plaintext Block

#### 3.2.1. Key Select

In this encryption, the key could be divided into two parts. (1) The information of selected gene sequence, as one part of the key, will be shared between sender and receiver through secure channel, but not available by the third party. In order to enhance the security of the algorithm, researcher sets start point and end point of the search position. The strength of this action is that we don't need to have full expression of a gene sequence, and the researchers would take encryption and decryption operations only from the position two sides agreed to start to the important position of the sequence of key and it obviously reduce expense. (2)The initial key of Key Generator, as another part of key, is used to control the Key Generator to create chaotic sequence. Because the Key Generator we designed is based on Logistic Mapping, in this part, the key are two parameters $\mu$ and $x_0$.

#### 3.2.2. Obtaining the Special Gene Sequence

From the GenBank （www.ncbi.nlm.nih.gov）, researcher could select chromosomal sequences as the mother sequence. For example, in this paper's simulation, the researcher selects the gene sequence of canis familiaris chromosome 1, and absolutely transmits it in security as a part of key.

### 3.2.3. Plaintext Block

Before encryption, the binary plaintext should be grouped together in 32-bit. For example, if the plaintext is "GENEGRYPTOGRAPHY", the result of its blocking is: Group1:"GENE", Group 2:"GRYP", Group 3:"TOGR", Group 4:"APHY". When the plaintext length can not be divisible by 4, the plaintext should be filled. The detail operation is: After Mod 4, we fill the remainder in the end of the plaintext. Such as, when result of the length of plaintext mod 4 is 3, we will fill three characters '3' in the end of plaintext.

## 3.3. Encryption

### 3.3.1. Encryption Process

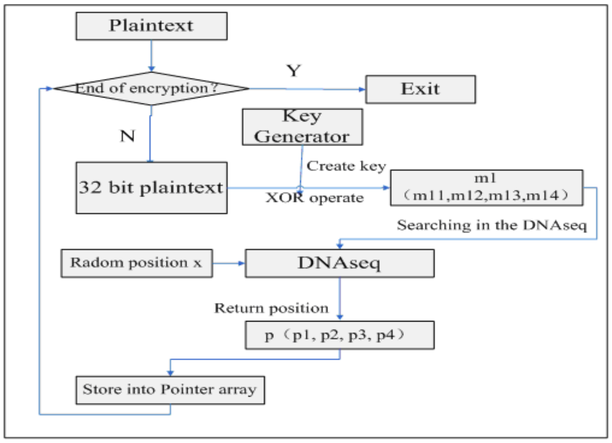

Figure 2. The Flow chart of Encryption

**Step1：** The researcher selects the Initial Key **key$_0$** randomly, which is composed by two double numbers, and then makes them to be the parameters of Logistic Mapping ($\mu$ and $x_0$); Besides, the Key Generator will create a 32-bit **key$_1$** to encrypt plaintext in the first encrypting; Then, researcher should transform plaintext into ASCII codes, and divide them into different groups; Finally, researcher should take XOR operation between the 32-bit plaintext and **key$_1$.** After these actions, the researcher will get the sequence **m$_1$**.

**Step2:** First, according to the DNA encoding presented in 3.1, **m$_1$** should be converted into DNA code. Through encoding, the researcher will gain the DNA sequence **se$_1$** which has 16-mer olio nucleotides; Secondly, based on the key and the information of DNA sequence which have been transmitted, the special DNA sequence (**DNAseq**) presented in them will be obtained; Then, researcher selects the position x (0<x<length of **DNAseq**) randomly, and from x the researcher searches the sequence to find the special string which is same as **m$_1$.** If there exist four continuous bases, **se$_{11}$**, **se$_{12}$**, **se$_{13}$**, **se$_{14}$** can totally be same as **m$_1$**. It will return the position of them: **p1**, **p2**, **p3**, **p4**; Finally, the algorithm restores these positions into the cipher-text array **Pointer**.

**Step3:** The Key Generator creates new encryption key **key$_n$** which will be used in the next process of encryption. And researcher should take XOR operation between 32-bit plaintext and **key$_n$**. Then, the new sequence **m$_n$** will appear.

**Step4:** Executing Step2 and Step3 without suspension until all plaintext's encryption having been completed, the program can be stopped.

The Decryption process is the reverse of the encryption process.

### 3.3.2. DNA Symmetric Encryption Cryptosystem

The Sender and the Receiver should transmit key and DNA sequence information through the secure channel. Sender sends cipher-text to receiver through public channel. When receiver gets cipher-text from sender, s/he will use DNA sequence information to get the right DNA sequence. Then receiver will use the key and the decryption algorithm to retrieve correct plaintext. The Encryption Cryptosystem is shown in figure 3.

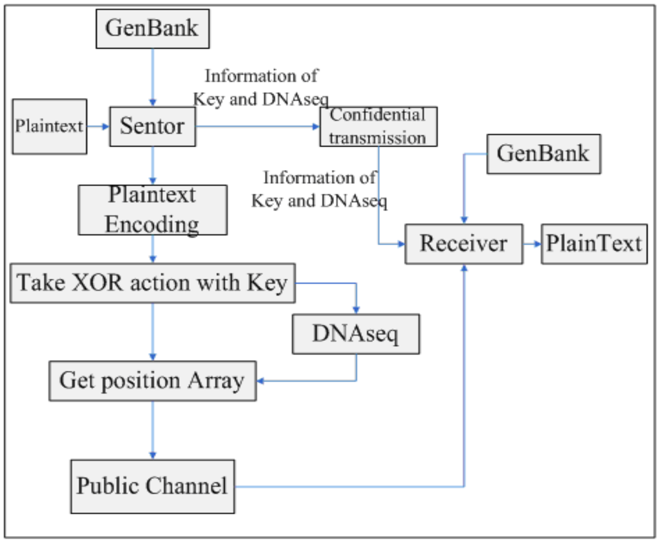

Figure 3. DNA Symmetric Encryption Cryptosystem

## 3.4. Key Generator

### 3.4.1. Logistic Mapping

One-Dimension Logistic mapping mathematical expression is as follows

$$x_{n+1} = \mu x_n(1-x_n)$$
$$n \in \{1,2,...\} \tag{4}$$

$0<=x_I<=1$, $\mu$ is the controlling parameter, $0<\mu<=4$

If $0<\mu<=1$, this problem will only have one solution 0, and no matter what the initial value $\mu$ selected, the final iteration will converge to 0.

If $0<\mu<=3$, this problem will have two solutions: 0 or1-1/$\mu$, after iterations, the solution converges to one of the two values.

If $3<\mu<=4$, the solution tends to chaos from period-doubling. Especially, if $3.5699456...<\mu<=4$, the solution looks towards chaos, and the value came from iteration will be a pseudo-random distribution status. If the $\mu$ tends to 4 extremely, the solution will be strong chaos.

### 3.4.2. Key Generator

1. Creating Chaos Sequence z: The value of Logistic

parameters $\mu$ and $x_0$ are serviced as key. Based on formula 4, the chaos sequence will be created:

$$z = (z_1, z_2, z_3, ..., z_k) \tag{5}$$

2. Creating Key: Each round, the key $z_i$ will be selected in the sequence $z$. According to formula (6), it will be converted to a 32-bit binary number:

$$[y_i]_2 = z_i \times 255 \tag{6}$$

Therefore, in every round, the key $Y$ will be expressed in formula (7):

$$Y = [y_i, y_{i+1}, y_{i+2}, y_{i+3}] \tag{7}$$

Explanation: chaos value $z_i$ came from the chaos sequence $z$. After multiplying 255, it will map to 8-bit binary $y_i$. Every four continuous $y_i$ will be used as one 32-bit key Y.

## IV. Analysis

### 4.1. Simulation Analysis

(1) Algorithm Validity

In the simulation designed by researchers, they encrypt the plaintext of "GENECRYPTOGRAPHY" (gene cryptography) and decrypt the cipher-text of "DNAENCRYPTION1" (DNA encryption). The results shown in table 1, demonstrate that the algorithm is correct.

TABLE I. The Result of the Simulation

| KEY | Key1 (0.5,3.9999) | Key2 (0.5,3.9999) |
|---|---|---|
| Plaintext | GENECRYPTOGRAPHY | DNAENCRYPTION1 |
| Cipher-text | 11843 12117 12080 12117<br>7912 8152 7900 7946<br>560 400 498 607<br>14189 14756 13945 13976 | 11590 11728 11746<br>11693 12047 11693<br>11675 13446 881<br>1158 1011 712 |
| Decrypted Plaintext | GENECRYPTOGRAPHY | DNAENCRYPTION1 |

(2) Key Space Analysis

The algorithm creates the key with the chaos mapping. The type of two parameters in the chaos mapping is double, thus the key space is huge. In table 2, the estimation of key space will be showed (The estimation's accuracy is $10^{-8}$ which is based on the computer-system's accuracy). Actually, with the contemporary society development, the accuracy of computer-system has surpassed $10^{-8}$. In other words, the key space will be larger.

TABLE II. Key Space Analysis

| | Status | Value Range | Key Space |
|---|---|---|---|
| Key1 (Para $x_0$) | Initial | (0, 1) | $1\times10^8$ |
| Key2 (Para μ) | Initial | (3.7, 4) | $0.3\times10^8$ |

(3) Key Sensitivity Analysis

When the researcher decrypts the cipher-text with the correct key, the plaintext will be retrieved.

When the researcher decrypts the cipher-text with the wrong key (0.5, 3.9998), the result of decryption is wrong.

When the researcher decrypts the cipher-text with the wrong key (0.4, 3.9999), the result of decryption is wrong.

Therefore, the algorithm is sensitive to the key.

TABLE III. Key Sensitivity Analysis

| Decrypt with key | Plaintext | Key($x_0$,μ ) | Decrypted Plaintext |
|---|---|---|---|
| Correct key | GENECRYP TOGRAPHY | (0.5, 3.9999) | GENECRY PTOGRAPHY |
| μ been changed | GENECRYP TOGRAPHY | (0.5, 3.9998) | 1QWsi0nX |
| $x_0$ been changed | GENECRYP TOGRAPHY | (0.4, 3.9999) | 87k.z#aa |

(4) Plaintext Sensitivity Analysis

In this simulation, the researcher measures the algorithm's sensitivity to plaintext. When the plaintext has been changed 1-bit, the Cipher-text is totally different with the plaintext. So, the algorithm is sensitive to the plaintext.

TABLE IV. Plaintext Sensitivity Analysis

| **Plaintext** | **Cipher-text** | **Key** |
|---|---|---|
| GENECRYP TOGRAPHY | 2410 5021 6639 4890<br>14240 12630 12678 12640<br>3448 3305 3415 3529<br>15542 13756 13808 14038 | (0.5, 3.9999) |
| GENECRYP TOGRAPHZ | 8408 8214 8372 8214<br>8581 8706 8902 8582<br>10810 10713 11009 11415<br>9821 12046 9863 10348 | (0.5, 3.9999) |
| GENECRYP TOGRAPHX | 13575 13945 13860 13945<br>3616 3747 3749 3632<br>3354 3414 3660 4332<br>1455 1286 1260 1237 | (0.5, 3.9999) |

### 4.2. Theoretical analysis

Because DNA Encryption System does not have a uniform model, scientists can't find an existing comprehensive standard to analysis the security of the algorithm. In this paper, the researcher analyses the security of the algorithm from the aspects of bio-security and math-security.

**1. Bio-Security**

As a result of that the encryption algorithm is designed to search the sequence on the special DNA which is same as the cipher-text, the attacker can't decrypt it without the information of the special DNA sequence. Without the key including the information of the special DNA sequence, it is impossible to find the special DNA sequence. If the attacker tries to decrypt the cipher-text by searching the DNA sequence exhaustively, there is no difference from the attacker to decrypt the cipher-text without key. Likewise, DNA has an extremely large data storage capacity, because one single chain of a chromosome has tens of millions of nucleotides. In order to find the correct starting position and end position, the attacker will spend numerous resources. In summary, the algorithm is safe enough from the aspect of bio-security.

**2. Math-Security**

In the simulation, the researcher selects the gene sequence of canis familiaris chromosome 1, whose length is 53004996 bp(base pair), as the mother sequence. It is no doubts that the key space could satisfy the all 8-bit binary coding. From the statistics, the number of each kinds nucleotides range from tens thousand to millions. And of course it decreases the possibility that encryption system will be attacked. If the researcher wants to achieve higher security, s/he could encrypt a longer DNA sequence. Besides, because of the relative action being operated on the DNA, the plaintext-oriented attack loss its meaning. In other words, the algorithm could assure the validity in preventing plaintext-oriented attack. In addition, the chaos mapping adopted in this algorithm, preprocesses the plaintext before the plaintext being converted into the position of DNA sequence. The Key Generator also makes assure that the key space is full of the capability to prevent the extensive attack. At the same time, the chaos mapping enhances the key's sensitivity. Even assume that the bio-security level had been broken, there is no doubt that the attacker still can't decrypt the cipher-text, because the attacker has to face the obstacle created by the math-security.

## V. Conclusion

In this paper, we originally proposed a symmetric-key cryptosystem based on the DNA symmetric cryptosystem and applying index. As a result of applying the block cipher, the cryptosystem can be standardized and synchronized completed easier. Besides, by applying the method that ciphers plaintext strings have a proper computing with the result come from pseudo-random number generator to create cipher-text, it presents a proper random key sequence to improve security. Through simulation and theoretical analysis, we can conclude that it provides an excellent performance of its encryption and an anti-attack ability; likewise, it has reached a calculated security level.

In future research, scientists should do further research to provide a theoretical proof of DNA cryptosystem's validity to make it be provable security level, and perfect the algorithm's security model. Also, they need to make full use of DNA computing and biological characteristics to eliminate the disadvantages of block cipher mode. For example, the data can not be hidden, or the Packet can not resist attacks of replacement, embedding and deleting.

In this paper, based on the endeavor of precursor, a meaningful exploration and research have made great development and innovation. It will help us enrich the content of DNA cryptography and emerge new ideas in this field.

## ACKNOWLEDGEMENTS

This work is supported by Aero-Science Fund of China (2009ZD53045), Science and Technology Development Project of Shaanxi Province Project(2010K06-22g), Basic research fund of Northwestern Polytechnical University (GAKY100101), and R Fund of College of Software and Microelectronics of Northwestern Polytechnical University (2010R001).

## References

[1] G ehani A, LaBean T H, Reif J H .D N A -based cryptography Dismacs Series in Discrete Mathematics and Theoretical Computer Science, 2000, 54: 233-249.

[2] Leier A, Richter C, Banzhaf W, et al. Cryptography with DNA binary strands. Biosystem s, 2000, 5(7): 113-22.

[3] Kazuo T, Akimitsu O, Isao S. Public-key system using D N A as a one-way function for key distribution.Biosystems, 2005, 81: 25-29.

[4] MX Lu, XJ Lai, GZ Xiao, L Qin, Symmetric-key cryptosystem with DNA technology, Science in China Series F: Information Sciences, 2007, 50(3), 324-333

[5] Kang Ning: A Pseudo DNA Cryptography Method CoRR 2009, abs/0903.2693

[6] Sherif T. Amin, Magdy Saeb, Salah El-Gindi, A DNA-based Implementation of YAEA Encryption Algorithm Proceedings of the Second IASTED International Conference on Computational Intelligence. 2006, 523: 32-36

[7] Celland C T, Risca V, Bancroft C. Hiding messages in DNA microdots. Nature, 1999, 399: 533—534

[8] CHEN Weichang, CHEN Zhihua. Digital Coding Of The Genetic Codons and DNA Sequences in High Dimension Space. A CTA BIOPHYSICA SINICA. 2000, 16(4), 760-768